基于条件生成对抗网络的漫画手绘图上色方法 *

梁培俊 [a]，刘怡俊 [a, b]

(广东工业大学 a. 计算机学院; b. 信息工程学院，广州 510006)

摘　要：对漫画手绘图进行上色在漫画和游戏开发领域是一项耗时耗力却又重要工作。因此，针对这项任务提出一种基于条件生成对抗网络(CGAN)的漫画手绘图自动上色方法。实验中，采用 U 型结构的生成器，对网络模型使用 L1 进行约束，在生成器和判别器的对抗式训练中，模型不断学习并优化手绘图到对应彩色图像间的映射关系，最后使用训练得到的条件 GAN 网络模型对手绘图上色。实验表明，使用这种方法可以有效并且快速地对漫画手绘图上色，同时保持可观的视觉效果。

关键词：漫画；手绘图；上色；深度学习；条件生成对抗网络

中图分类号：TP183　　doi: 10.3969/j.issn.1001-3695.2017.07.0891

Colorization of manga sketch based on conditional generative adversarial networks

Liang Peijun[a], Liu Yijun[a, b]

(*a. College of Computer*, *b. College of Information Engineering*, *College of Computer Guangdong University of Technology*, *Guangzhou* 510006, *China*)

Abstract: Colorization of manga sketch is an time-consuming and labor-intensive but significant task in both manga and game development. Therefore, for this task, this paper proposed a method to color manga sketch in unsupervised based conditional generative adversarial network(CGAN) . In the experiments, adopting a generator with an U-Net structure, constraining the model with L1 term, in the adversarial training between the generator and the discriminator, model continuously learns and optimizes the mapping from manga sketch to its corresponding colorful image. At last, GAN that generated model from training could be used to color manga sketch. Experiments results are shown to demonstrate the effectiveness of rapid colorization for manga sketch as well as the plausibility of visual effects.

Key Words: manga; sketch; colorization; deep learning; conditional generative adversarial network(CGAN)

0　引言

随着漫画和数字媒体产业的蓬勃发展，各种漫画产品出现的速度也在逐步加快，人们对于漫画产品的需求也不再只满足于黑白的线条画面上。但是由于一幅漫画生成是由画家从黑白手绘图开始制作，然后经过一系列繁杂处理之后最后才是上色。这个过程中，无论是手绘还是上色步骤都是十分耗时耗力。正是因为这种工作的具有时间和人力密集的性质，所以直到现在市面上很多的漫画产品也只有封面或者前几页才是彩色画面，并借此来吸引买家的注意力。因此，如果借助深度学习的技术，可以将上色这一过程无监督化，将会在时间和效率上给漫画的制作带来质的提高。

文献[1～3]之前的图像上色方法主要依赖于灰度区域的连续性来找到灰度相同的区域，从而辅助于对图像的分割再进行彩色化。然而黑白的手绘图只存在图像轮廓线条，并不像灰度图像那样有着连续且有级别差异的灰度区域可以被用于图像分割，因此灰度图像彩色化的方法并不能有效正常地对手绘图进行上色处理，从图 1 灰度上色效果可知。

Gatys 等人在文献[4～6]中提出一种能够进行艺术风格转换的方法，这种方法的关键是利用卷积神经网络中图像的内容和风格的特征可分性，将 2 张不同图片中各自取出一种特征，再将抽取出来的 2 中特征结合起来生成一种全新的图片，其内容和风格来自 2 张不同的图片。但是，如果使用这种模型来完成漫画手绘图上色并不能达到正常的上色功能。因为风格并不等于颜色这一属性，同样内容也不是图像轮廓这以属性，从图 1 风格转换效果可知。

在 2014 年等人[7]提出的开创性的生成对抗网络（GAN），该模型由生成器模型和判别器模型两部分组成，生成器模型主要用于捕获样本的数据分布，而判别器模型承担的是二分类器的角色，用于判别输入数据是有生成器模型产生的抑或是真实

基金项目：广东省和广州市科技计划项目（201604010051，2015B090901060，2016B090903001，2016B090904001，2016B090918126，2016KZ010101）

作者简介：梁培俊（1993-），男，广东汕尾人，硕士，主要研究方向为深度学习（junecthe@126.com）；刘怡俊（1977-），男，教授，硕导，博士，主要研究方向为集成电路设计、深度学习、类脑计算机、北斗导航芯片、物联网.

数据。模型的优化过程简单来说就是，由生成器模型生成一张图像来欺骗判别器模型，接着由判别器模型去判别这张图片的真实性，然后整个网络迭代地将这两个模型进行对抗训练，随着训练的过程的进行，两个模型的能力越来越强，最终达到稳态。借助于这种博弈式的优化，生成对抗网络能够生成高质量的图像，然而用这种网络的训练方式太过自由，如果使用这种方法对手绘图进行训练后染色会产生畸形的结果，比如相同的颜色会被填充于不相关的区域。



图 1　上色方法效果比较

漫画手绘图的上色需要考虑两个关键点：生成的彩色图像与原本的手绘图的底层轮廓信息不变，即底层信息不变；相同的颜色不可以填充到不相关区域，即上色合理性。本文在充分考虑底层信息的不变性和上色合理性的基础上，构建了一种用于漫画手绘图上色的条件生成式对抗网络[8]。该网络包含生成器模型和判别器模型，其中生成器模型使用了带有跳跃连接的U 型[9]解码器－编码器[10]网络结构。U 型网络结构如图 2 所示。在 U 型网络下底层的信息可以直接达到解码器部分，从而达到底层信息的不变性。

判别器模型以块的方式（PatchGAN）[11]判别图像是否真实，对每张生成彩色手绘图像以 $N \times N$ 大小的块来判别，最后以其均值作为结果。此外网络的目标函数加入了 L1 约束项，使得生成的图像更加清晰。该网络以黑白的漫画手绘图作为生成器模型的输入，在生成器模型的解码器部分以 Dropout 的方式加入噪声，以生成的彩色漫画图像作为输出，然后与判别器模型以对抗式的方式进行训练，不断优化彩色图像与手绘图之间的映射关系，最终得到趋于稳态的网络模型。本文的训练和测试所用的数据集是从 safebooru 上抓取并加工处理得到的，使用 Pytorch 在 GPU 上训练网络参数。最后使用得到的网络模型对手绘图无监督上色，从图 1 本文方法效果可知。

1　CGAN 漫画手绘图上色网络

1.1　CGAN 原理

GAN 是一种生成模型，包含生成器模型 G 和判别器模型 D，将从随机噪声向量 z 到输出图像 y 的匹配关系定义为 $G: z \rightarrow y$，生成对抗网络对这种匹配关系进行学习。相比之下，CGAN 学习的关系是从被观测图像 x 和随机噪声向量 z 到输出图像 y 的匹配关系，这里定义为 $\{x, z\} \rightarrow y$。生成器 G 被训练用于生成输出结果，并且使得判别器模型 D 判断该输出结果是真实图像，从而使其生成的图像效果不断逼近与真实的图像。而判别器模型用于检测由生成器模型生成的图像。训练过程如图 3 所示。对生成器模型 G 输入条件信息黑白手绘图和随机噪声向量 z，然后生成彩色图像。而判别器模型 D 使用生成器模型 G 生成的图像和原始的彩色图像进行训练。D 的目标是为了区分从生成图像中判别出真实图像。两个网络以对抗的方式不断优化训练，构建出手绘图像到彩色图像间的映射关系，最终使网络具有无监督上色的功能。

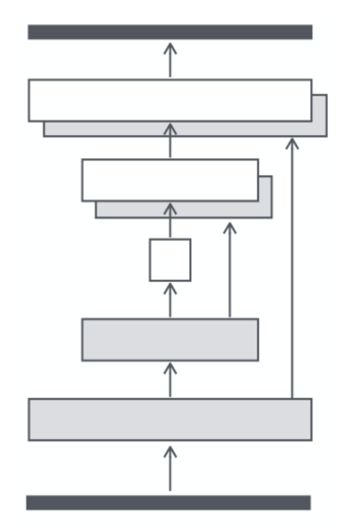


图 2　U 型网络结构

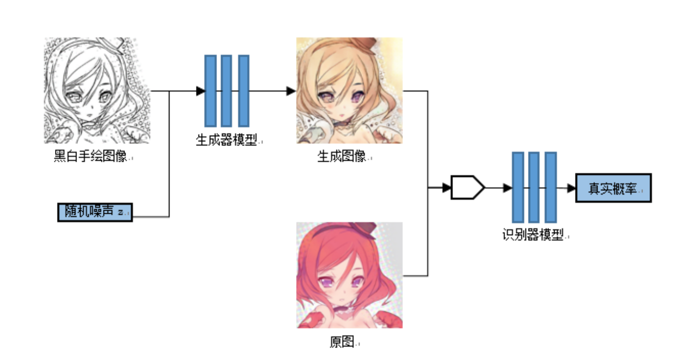


图 3　CGAN 漫画手绘图上色网络训练过程

生成对抗网络的目标函数可以表示为

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y \sim p_{data}(x,y)}[\log D(x,y)] + \mathbb{E}_{x \sim p_{data}(x), z \sim p_z(z)}[\log(1 - D(x, G(x,z)))] \tag{1}$$

生成器模型尽力最小化目标函数，判别器模型尽量最大化它，即

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) \tag{2}$$

为了测试对约束判别器的重要性，这里介绍一种判别器不

对输入图像 x 进行观测的版本：

$$\mathfrak{L}_{cGAN}(G,D)=\mathbb{E}_{x,y\sim p_{data}(y)}[\log D(y)]+\mathbb{E}_{x\sim p_{data}(x),z\sim p_z(z)}[\log(1-D(G(x,z))] \quad (3)$$

之前的条件式生成对抗网络已经发现了将 GAN 目标函数与传统的损失函数如 L2 距离[12]结合更有利。判别器模型的任务保持不变，但是生成器模型的任务不仅是欺骗判别器模型，而且要在 L2 的基础上贴近真实输出。使用 L1 距离代替 L2，因为 L1 产生结果更为清晰，此时目标函数如下：

$$\mathfrak{L}_{L1}(G)=\mathbb{E}_{x,y\sim p_{data}(x,y),z\sim p_z(z)}[\| y-G(x,z)\|_1] \quad (4)$$

最终的目标函数为

$$G^*=\arg\min_G\max_D \mathfrak{L}_{cGAN}(G,D)+\lambda\mathfrak{L}_{L1}(G) \quad (5)$$

在没有随机噪声 z 的情况下，网络仍然可以学习到从 x 到 y 的匹配关系，但是会生成确定性的输出，并因此无法匹配到除 delta 函数之外的分布。过去的条件生成对抗网络已经意识到了这一点并因此提供高斯噪声 z 和输入图像 x 作为生成器模型的输入。这里以 dropout 的形式提供噪声，并在训练和测试中都将其应用在生成器模型的一些层中。这样模型能够产生随机输出并因此捕获到模型产生的所有熵。最后在网络的输出中，除了 dropout 噪声之外，也可以观察到少量的随机性。

1.2 网络架构与实现详情

本文采用模型包括生成器模型和判别器模型。用 Ck 表示一个带有 k 个卷积核的层，按照卷积计算、批正规化和 ReLU 激活函数的顺序[13]处理输入信息。CDk 表示带有 dropout 率为 0.5 的层，该层按照卷积计算、批正规化、Dropout 和 ReLU 激活的顺序处理输入数据。所有卷积计算中使用 4×4 的卷积核，2 的步长。编码器和判别器模型中的卷积计算的下采样因子为 2，而在解码器中，卷积计算的上采样因子为 2。

1.2.1 生成器模型架构

漫画手绘图上色问题的一个确定特性是它们将高分辨率的输入网格与高分辨率的输出网格进行匹配。另外，输入和输出虽然在颜色表现上有所不同，但是效果图都有相同的内在结构即相同的轮廓。因此，输入的结构和和输出的结构几乎对齐。因此生成器模型的架构也围绕这些特点进行设计。

在图像彩色化的问题上，有许多解决方案都是采用一个编码器-解码器网络，如图 4 所示，在这样的网络中的编码器部分，通过每一层时会对输入进一步下采样。这种网络设计要求所有信息流过所有的层包括瓶颈层。对于图像上色这类问题，在输入和输出之间有大量的底层信息被共享，比如，手绘图上色的情况中，输入和输出图像共享着突出边缘的位置信息，因此可取的做法是让这些信息可以直接通过网络。

为了让需要共享的信息可以在生成器模型中能够绕过瓶颈层，需要在生成器模型中加入跳跃连接，设计成 U 型结构。图 5 为生成器模型结构。

生成器模型包含编码器部分和解码器部分。其中 C64-C128-C256-C512-C512-C512-C512-C512 为编码器部分，而 CD512-CD1024-CD1024-C1024-C1024-C512-C256-C128 为解码器部分。在解码器的最后一层之后是一个卷积计算，用来匹配输出通道，接着是一个 tanh 函数。上述提到的要求需要做一个特殊处理，批正规化处理没有应用在编码器的 C64 层。编码器中使用的所有 ReLU 激活函数类型都是斜率为 0.2 的 LeakyReLU。而解码器中的是 ReLU 激活函数。

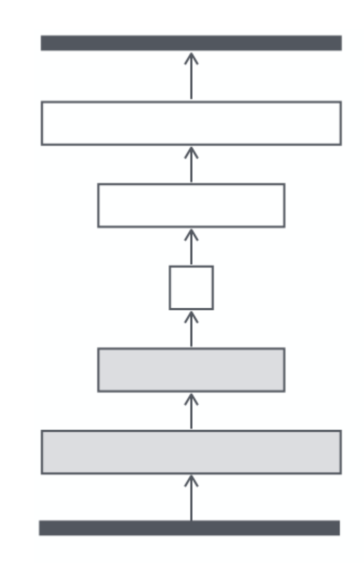


图 4 编码器-解码器网络

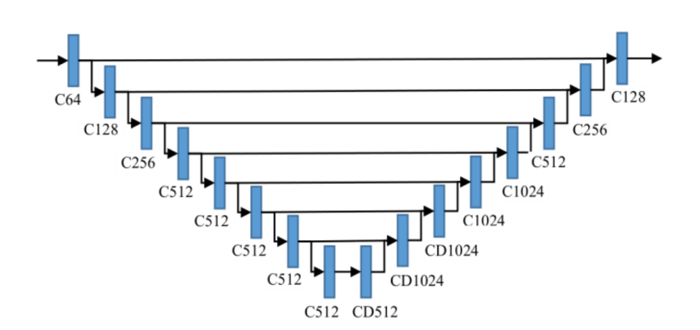


图 5 生成器模型结构

设 n 是总层数，除了编码器中的每个第 i 层与解码器中的第 $n\text{-}i$ 层之间的跳跃连接，U 型网架构是相同的。跳跃连接将第 i 层到第 $n\text{-}i$ 层的激活函数连接起来同时改变了解码器中的通道数量。

1.2.2 判别器模型结构

如式（5）所示，依赖 L1 项加强低频正确性，加强判别器只对高频结构进行建模的约束，增加生成的彩色图像的清晰度。此外，为了对高频建模，判别器模型结构采用了 PatchGAN，这样能够只以块的规模来惩罚结构。判别器模型尽力地区分每张图像中 $N\times N$ 大小的每一块是否真实，然后平均所有响应作为最终输出。

图 6 为判别器型结构。在最后一层后面是一个卷积计算，用于匹配一维输出，接下来是 Sigmoid 函数。需要特殊处理的是，C64 层不使用批正规化处理。判别器模型中使用的所有 ReLU 激活函数类型都是斜率为 0.2 的 LeakyReLU。

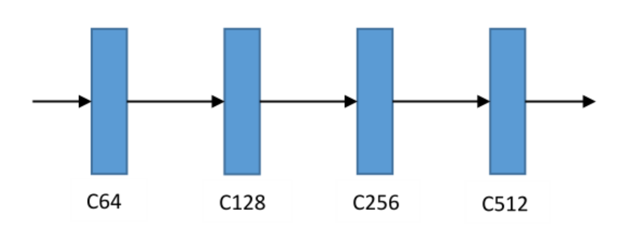


图 6　判别器模型结构

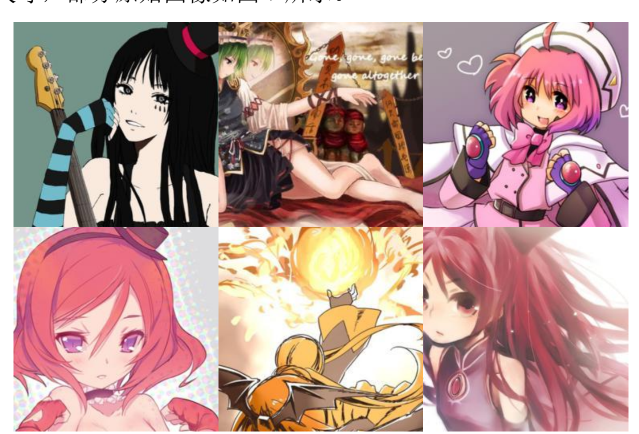
2　实验

2.1　实验平台

训练过程中卷积计算需要大量的矩阵运算，图形处理单元（GPU）比中央处理单元（CPU）能够极大地加速训练速度，缩减训练时间，因此使用 GPU 更为合适。本文实验平台为 Ubuntu16 64 位操作系统，Intel CPU i5，内存 8 GB，GPU 使用 NVIDIA GTX 1050Ti，同时使用 Pytorch 深度学习框架进行实验。

2.2　实验数据与处理

本文训练所采用的漫画图像数据集使用网络爬虫从著名的漫画素材网站 safebooru 上随机抓取，并经过筛选剔除了不适合用于训练的数据图像，最终剩下的漫画图像数量为大约 2.7 万张。然后使用 Imagemagick 对图像进行批量缩放到 255×255 的尺寸，部分原始图像如图 7 所示。

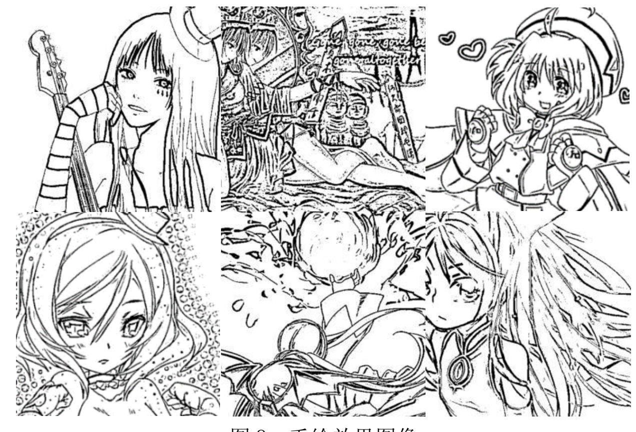

图 7　部分原始图像

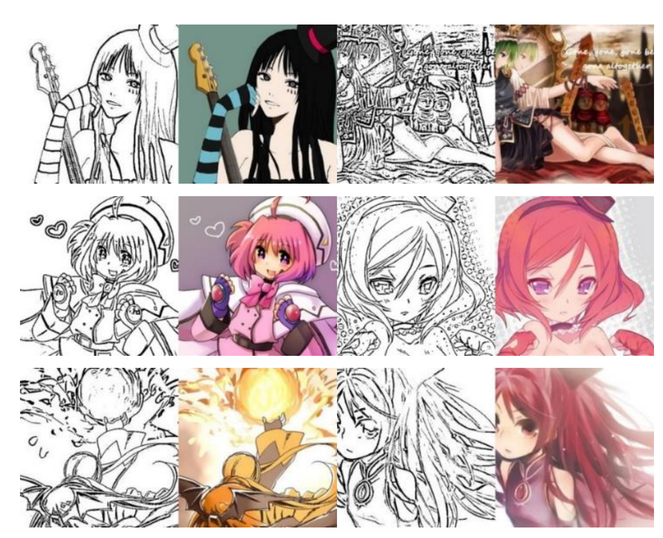
再使用 opencv 将所有漫画图像进行批量地轮廓提取处理，使其转换为漫画手绘效果图。手绘效果图像如图 8 所示。

接着取前 3 000 张图像作为测试数据图像，剩下的作为训练数据图像。为了使图像数据更能易于读取和比较，这里还对原图像和对应的处理后的图像一一连接起来，最终用于训练和测试，如图 9 所示。

2.3　模型训练

训练过程中，在判别器模型和生成器模型间交替地执行梯度下降，使用最小批量的随机梯度下降算法和 Adam 求解器[14]。对于批量的大小设置为 1，在文献[15]中已经证明该设置在图像生成任务上的有效性。实验中的学习率设置为 0.000 2，迭代次数为 200，训练时间需要 3 天，训练过程用到 visdom 服务器来可视化训练效果。


图 8　手绘效果图像


图 9　训练图像数据

2.4　实验结果

为了评估训练模型是否可以用于对黑白漫画手绘图上色，本文使用训练好的模型对测试集中的 3 000 张手绘图像进行上色测试，平均一张漫画手绘图上色耗费时间约 0.8 s。无监督上色测试结果如图 10 所示。可知，本文提出的网络模型的对漫画手绘图的上色效果满足了底层信息不变和上色合理性两个关键点，且整体效果保持较为清晰和合理。但是存在上色效果图存在模糊的地方，颜色方面较为单一的瑕疵。

除了验证模型对漫画手绘图上色能力外，还对其进行了自然灰度图像上色能力进行实验。这部分训练和测试数据来自 ImageNet 的狗类图像，数量分别为 1 万和 1 千张。预处理和训练方式与漫画手绘图实验部分类似，上色效果如图 11 所示。由上色效果可看出，该模型对自然灰度图像的有一定可行性，但存在部分细节颜色误填的情况。

3　结束语

本文提出了一种基于条件 GAN 的漫画手绘图无监督上色方法，构建了动漫手绘图上色的条件对抗生成网络模型，通过对目标函数添加 L1 约束提高上色效果图的清晰度，采用 U 型结构的生成器模型来保留手绘图与彩色图的底层轮廓信息，使

用生成器和判别器对抗式地训练，构建手绘图与彩色图间的映射关系，最终产生可以用于对一般黑白手绘图进行上色的模型，从而实现黑白漫画手绘图的无监督上色。实验表明，这种方法能够对手绘图进行合理地上色同时保持一定清晰度。后续将对上色效果的清晰度和配色方面进行进一步研究。



图 10　无监督上色测试结果

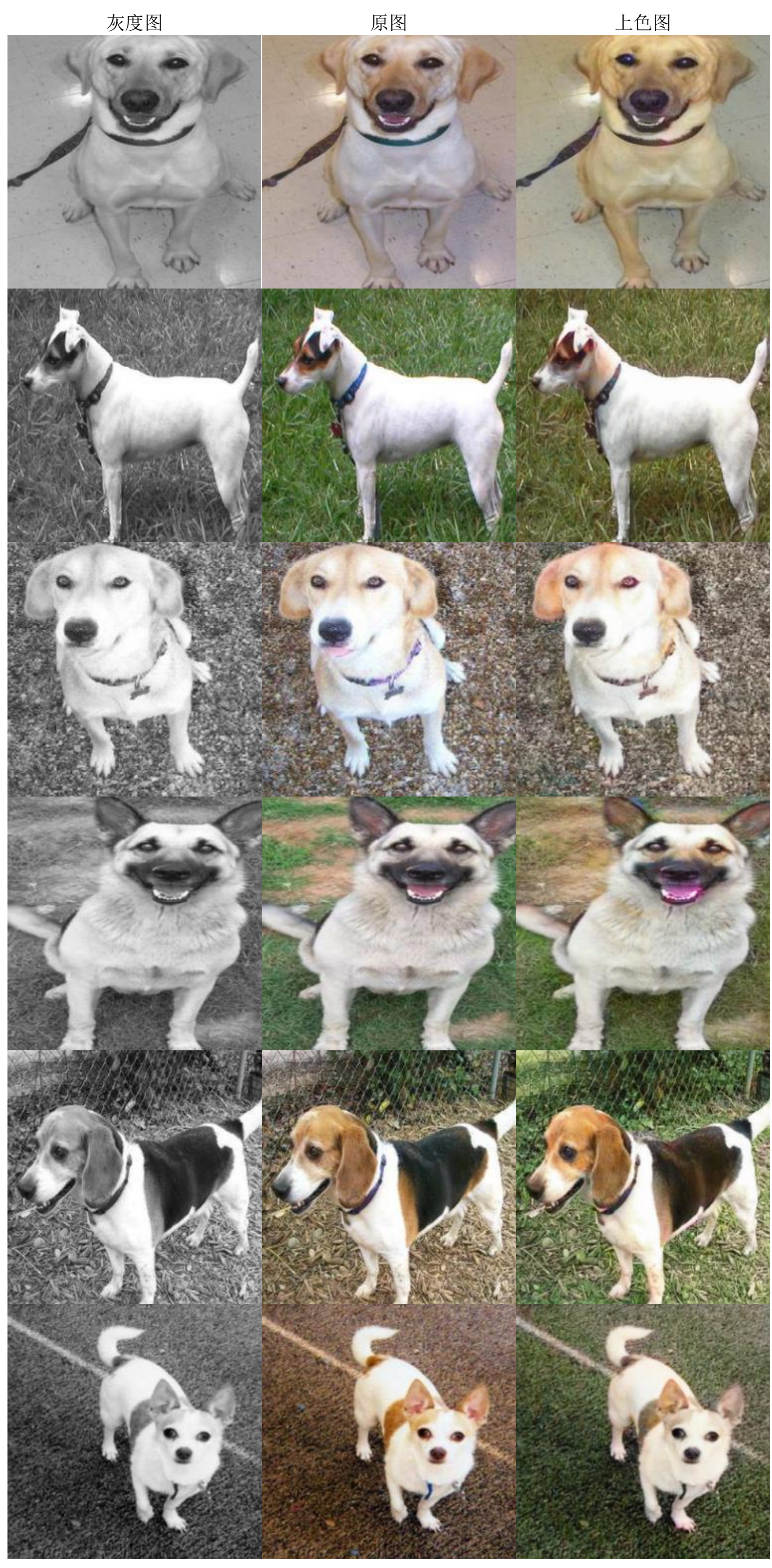


图 11 灰度图像上色结果

参考文献：

[1] Zhang R, Isola P, Efros A A. Colorful image colorization [C]// Proc of European Conference on Computer Vision. [S. l.] : Springer International Publishing, 2016: 649-666.

[2] Levin A, Lischinski D, Weiss Y. Colorization using optimization [J]. ACM

Trans on Graphics, 2004, 23 (3): 689-694.

[3] Cao Yun, Zhou Zhiming, Zhang Weinan, et al. Unsupervised diverse colorization via generative adversarial networks [J]. arXiv preprint arXiv: 1702. 06674, 2017.

[4] Gatys L A, Ecker A S, Bethge M. A neural algorithm of artistic style [J]. arXiv preprint arXiv: 1508. 06576, 2015.

[5] Gatys L A, Ecker A S, Bethge M. Image style transfer using convolutional neural networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. [S. l.] : IEEE Computer Society, 2016: 2414-2423.

[6] Johnson J, Alahi A, Li F F. Perceptual losses for real-time style transfer and super-resolution [C]// Proc of European Conference on Computer Vision. 2016: 694-711.

[7] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets [C]// Advances in Neural Information Processing Systems. 2014: 2672-2680.

[8] Mirza M, Osindero S. Conditional generative adversarial nets [J]. arXiv preprint arXiv: 1411. 1784, 2014.

[9] Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation [C]// Proc of International Conference on Medical Image Computing and Computer-Assisted Intervention. 2015: 234-241.

[10] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks [J]. Science, 2006, 313 (5786): 504-507.

[11] Li C, Wand M. Precomputed real-time texture synthesis with Markovian generative adversarial networks [C]// Proc of European Conference on Computer Vision. [S. l.] : Springer International Publishing, 2016: 702-716.

[12] Pathak D, Krahenbuhl P, Donahue J, et al. Context encoders: feature learning by inpainting [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2536-2544.

[13] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift [C]// Proc of International Conference on Machine Learning. 2015: 448-456.

[14] Kingma D, Ba J. Adam: a method for stochastic optimization [J]. arXiv preprint arXiv: 1412. 6980, 2014.

[15] Ulyanov D, Vedaldi A, Lempitsky V. Instance normalization: the missing ingredient for fast stylization [J]. arXiv preprint arXiv: 1607. 08022, 2016.